Computer Networks

Dr. Ali Sayyed
Department of Computer Science
National University of Computer & Emerging Sciences

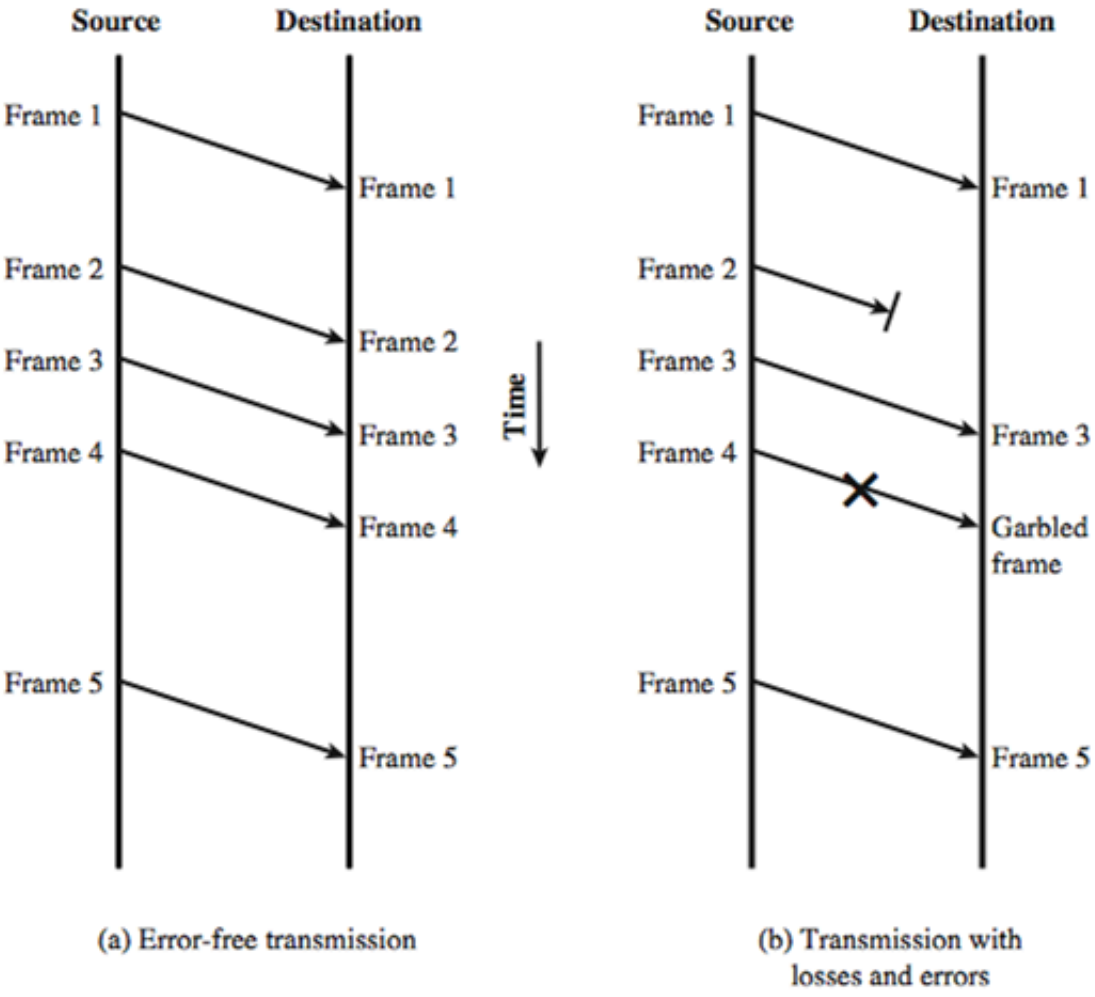
Data Link Layer

Flow Control

Flow Control

- Ensure sending entity does not overwhelm receiving entity
  - —by preventing buffer overflow
- Upon receiving, the receiver must do some processing before passing the data to the higher-level software.
- Each transmitted frame suffers an arbitrary and variable amount of delay:
  - —transmission time
    - time taken to emit all bits into medium
  - —propagation time
    - time for a bit to traverse the link

Model of Frame Transmission



(a) Error-free transmission

(b) Transmission with losses and errors

Flow Control - Stop and Wait

- The simplest form of flow control,
  - —source transmits frame
  - —destination receives frame and replies with acknowledgement (ACK)
  - —source waits for ACK before sending next
  - —destination can stop flow by not send ACK
  - —works well for a few large frames
  - —Inadequate if large block of data is split into small frames
    - In most cases, a large block of data is broken into smaller blocks (because of limited buffer size, errors detected sooner, easy retransmission etc).

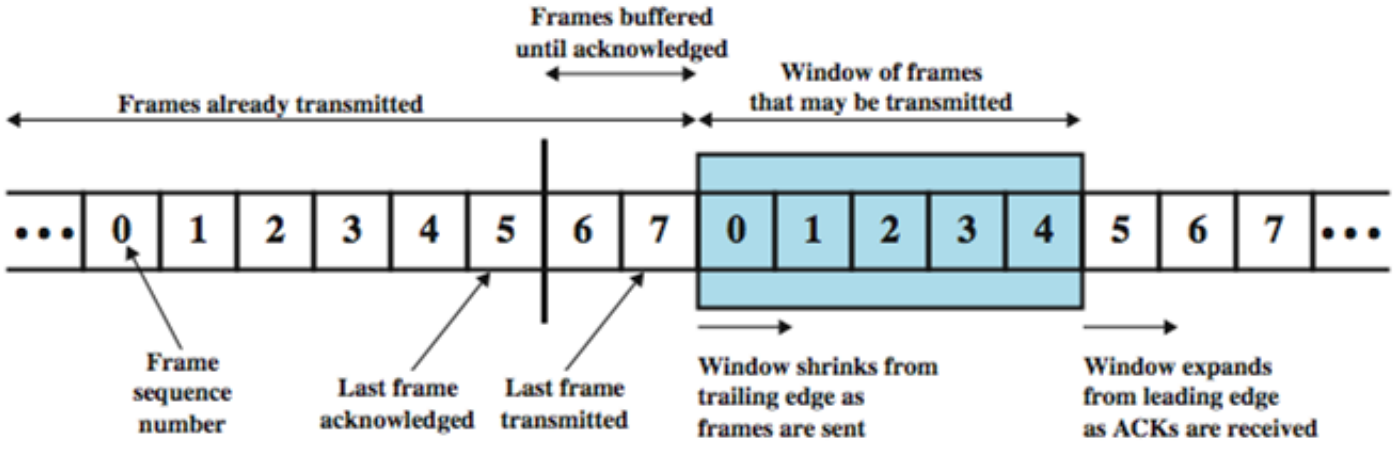
Stop and Wait Link Utilization

- The stop-and-wait procedure may be inadequate, mainly since only one frame at a time can be in transit
- The communication link is always underutilized or inefficiently utilized.
- In essence, for very high data rates, for very long distances between sender and receiver, stop-and-wait flow control is inefficient.

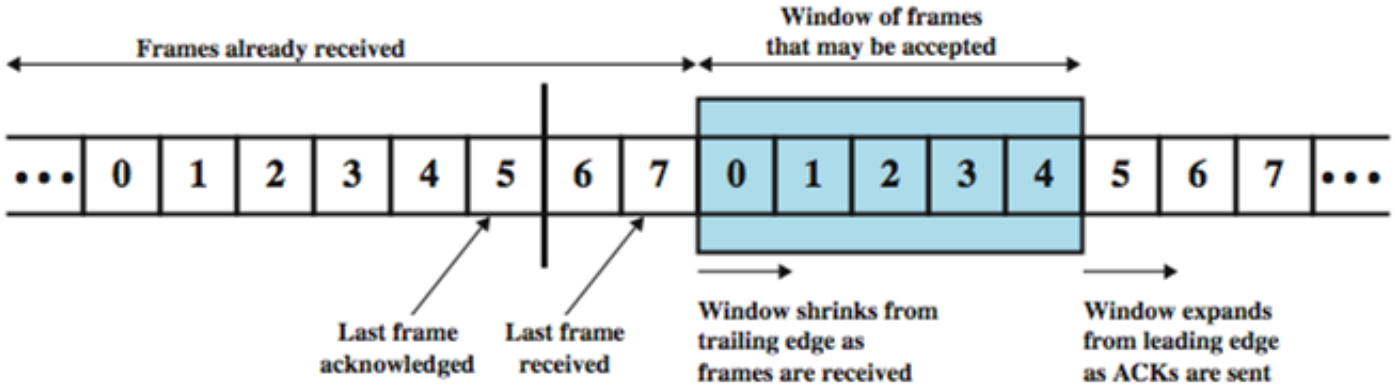
Flow Control - Sliding Windows

- Allows multiple numbered frames to be in transit
- Consider, Receiver has a buffer with **W** capacity
- Transmitter sends up to **W** frames without ACK, assuming that the receiver can handle it.
- ACK includes seq number of next frame expected, hence can also be used to acknowledge multiple frames

Sliding Window Diagram

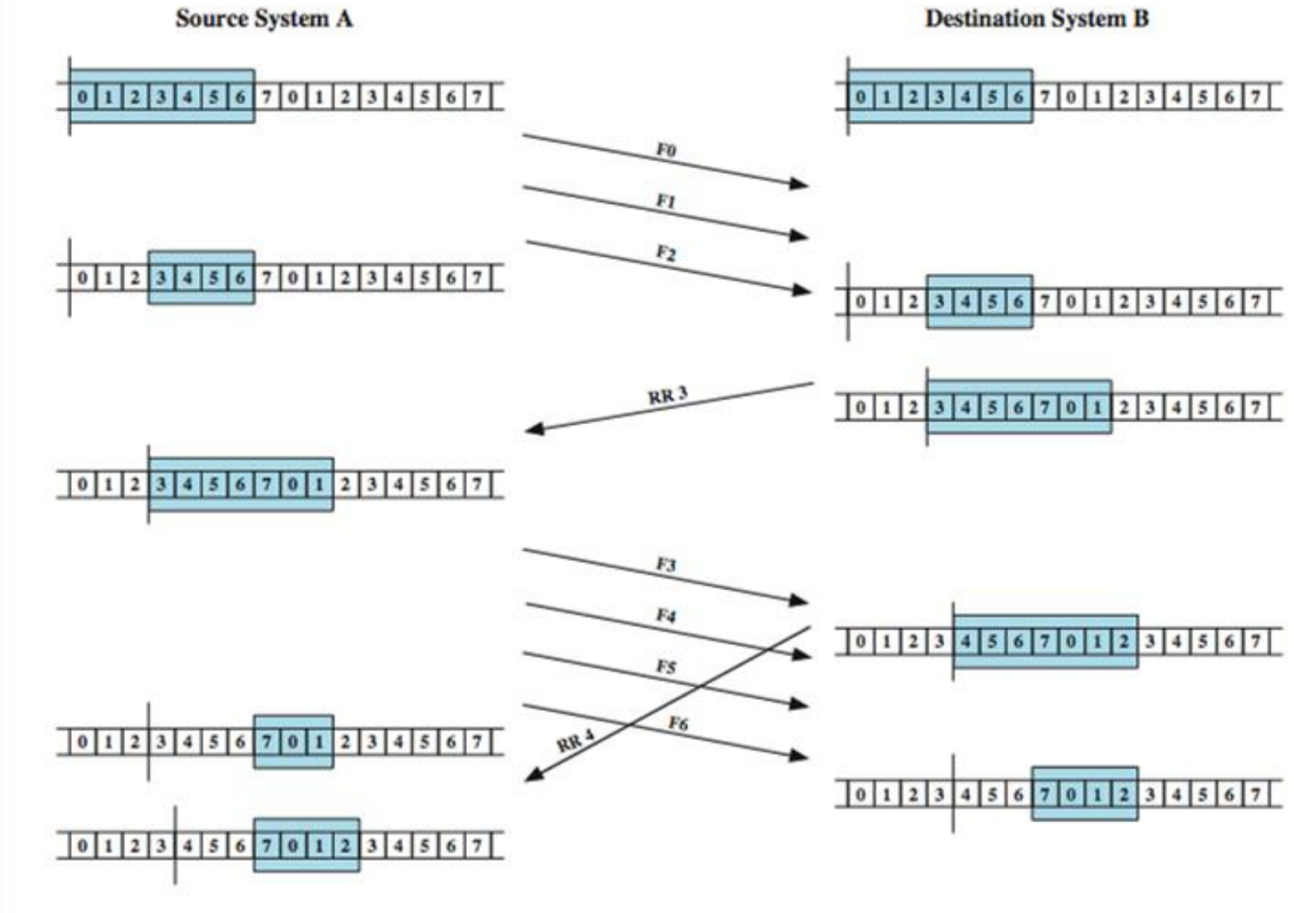


(a) Sender's perspective



(b) Receiver's perspective

Sliding Window Example



Error Control

Error Control

- Error control refers to mechanisms to detect and correct errors that occur in the transmission of frames.
- In addition, we admit the possibility of two types of errors:
  - **Lost Frame:** frame fails to arrive at the other side, eg. because a noise burst damaged a frame so badly it is not recognized
  - **Damaged Frame:** A recognizable frame does arrive, but some of the bits are in error (have been altered during transmission)

Lost Frames - Automatic Repeat Request (ARQ)

- Collectively, the mechanisms used for lost frames are all referred to as automatic repeat request (ARQ).

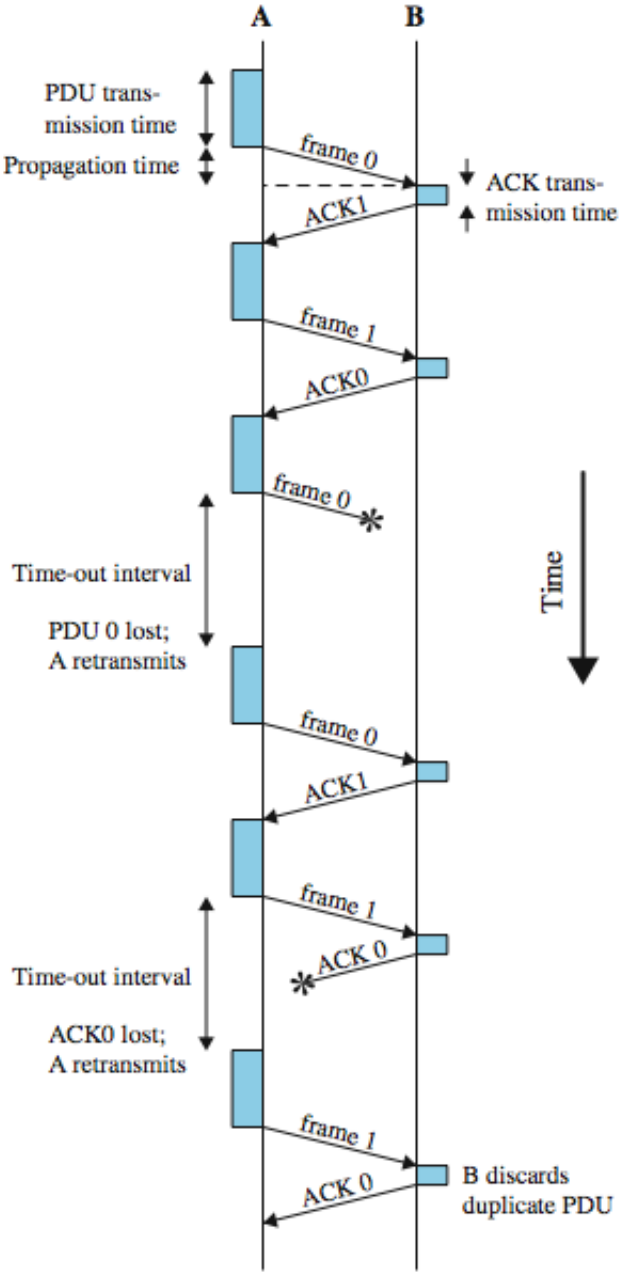
- The effect of ARQ is to turn an unreliable data link into a reliable one.
- Three versions of ARQ have been standardized:
  - Stop-and-wait ARQ
  - Sliding Window ARQ
    - Go-back-N ARQ
    - Selective-reject ARQ

Stop and Wait

- Based on the stop-and-wait flow control technique outlined previously.
- Source transmits single frame and wait for ACK
- Two types of errors could occur
  - —If received frame damaged, discard it
    - if transmitter has no ACK within timeout, retransmit
  - —If ACK damaged, transmitter will not recognize it
    - transmitter will retransmit
    - receive gets two copies of frame but can discard one

Stop and Wait

- see example with both types of errors
- pros and cons
  - simple
  - inefficient



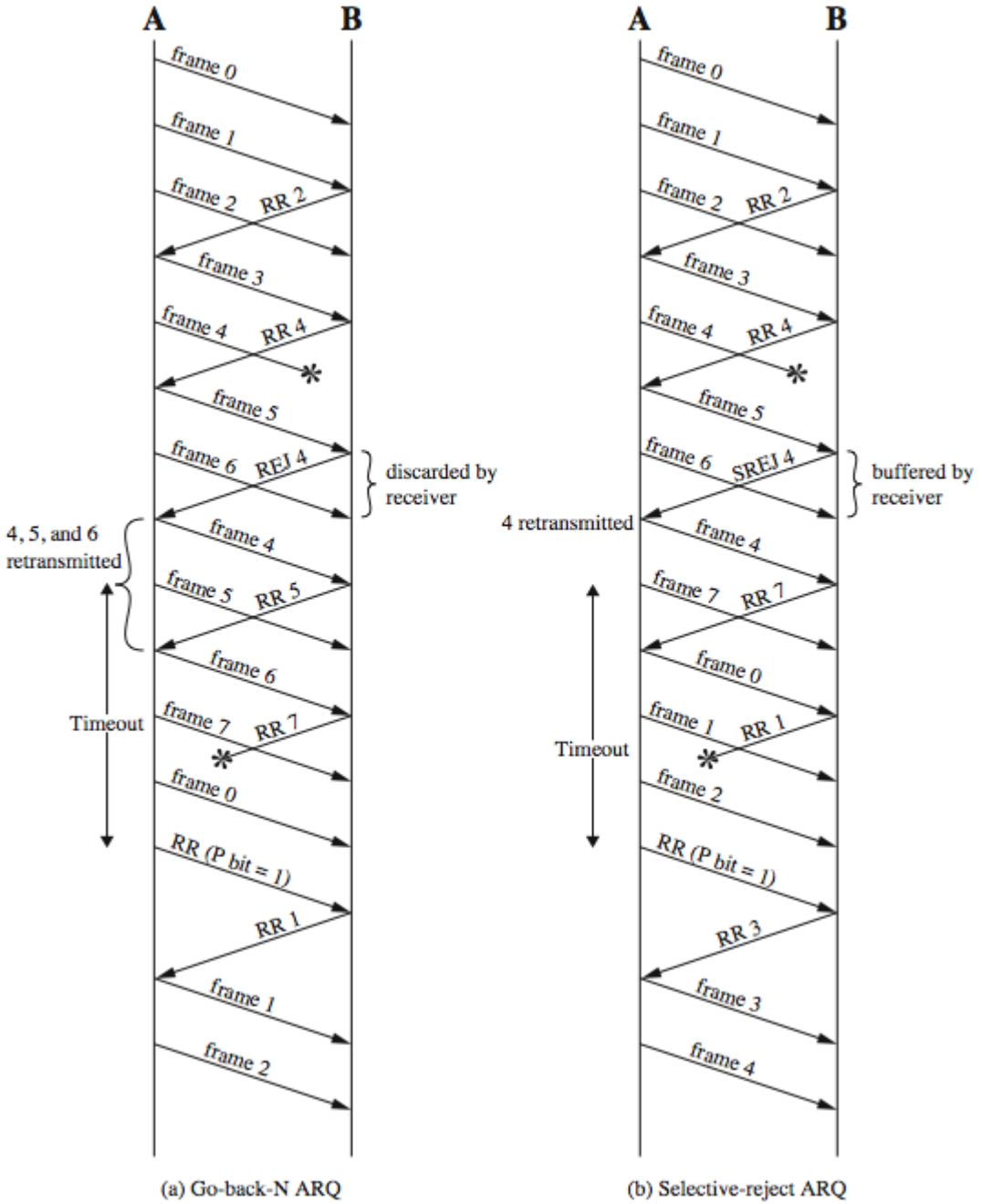
Go Back N

- based on sliding window
- if no error, ACK as usual
- use window to control number of outstanding frames
- if error, reply with rejection
  - discard that frame and all future frames until error frame received correctly
  - transmitter must go back and retransmit that frame and all subsequent frames

Selective Reject

- also called selective retransmission
- only rejected frames are retransmitted
- subsequent frames are accepted by the receiver and buffered
- minimizes retransmission
- receiver must maintain large enough buffer
- more complex logic in transmitter
- hence less widely used
- useful for satellite links with long propagation delays

Go Back N vs Selective Reject



Error Control – Damaged Frames

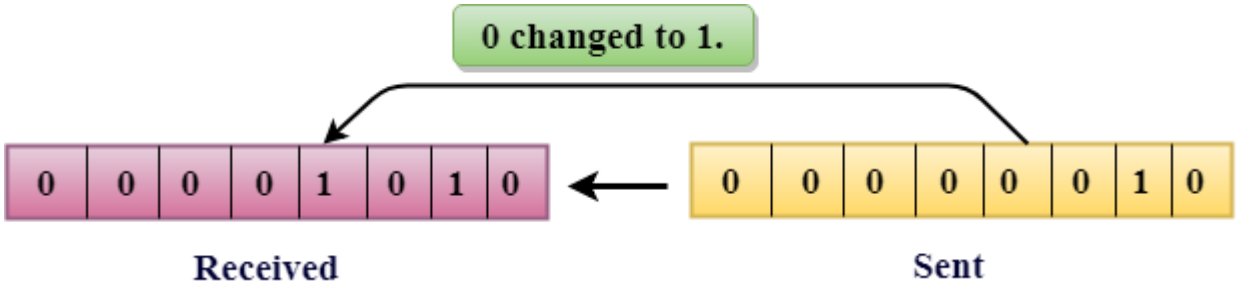
- The most common error control method is to compute and append some form of a checksum to each outgoing frame at the sender's data link layer and to recompute the checksum and verify it with the received checksum at the receiver's side. If both of them match, then the frame is correctly received; else it is erroneous. The checksums may be of two types:
  - **Error detecting** : Receiver can only detect the error in the frame and inform the sender about it.
  - **Error detecting and correcting** : The receiver can not only detect the error but also correct it.

Types Of Errors

- Errors can be classified into two categories:
  - Single-Bit Error
  - Burst Error

Single-Bit Error

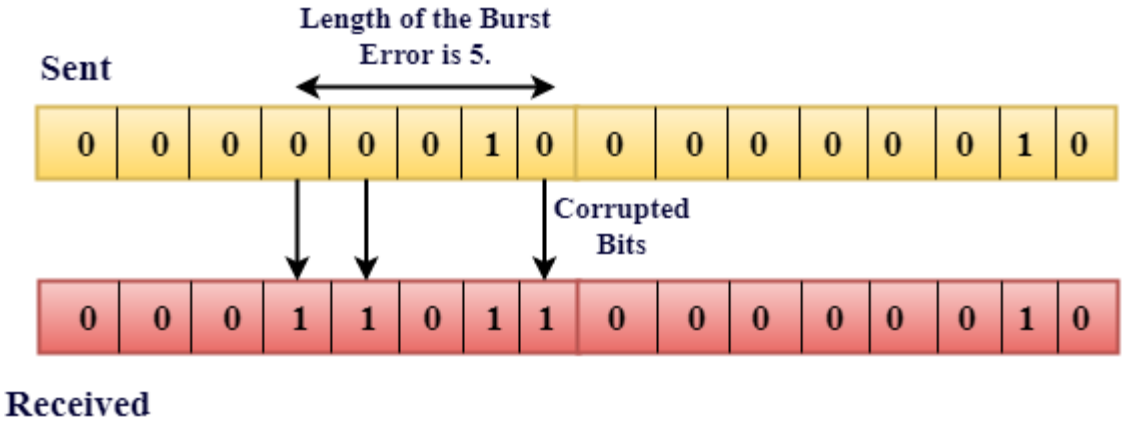
- One bit of a given data unit is changed from 1 to 0 or from 0 to 1.



- Single-Bit Error does not appear more likely in Serial Data Transmission.
  - For example, if a sender sends the data at 10 Mbps the noise must have a very short duration which is very rare.
  - However this kind of errors can happen in parallel transmission

Burst Error

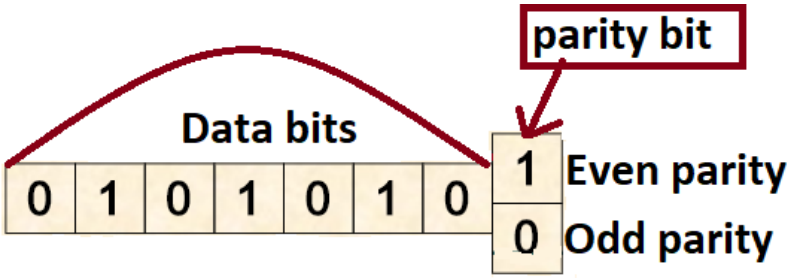
- two or more bits are changed from 0 to 1 or from 1 to 0 is known as Burst Error.
- The Burst Error is determined from the first corrupted bit to the last corrupted bit.



- Burst Errors are most likely to occurr in Serial Data Transmission.

Error Detection – Single Parity Bit

- Simple example of error detection technique is parity bit. The parity bit is chosen such that the number of 1 bits in the code word is either even(for even parity) or odd (for odd parity).
- For example when 10110101 is transmitted then for even parity an 1 will be appended to the data and for odd parity a 0 will be appended.

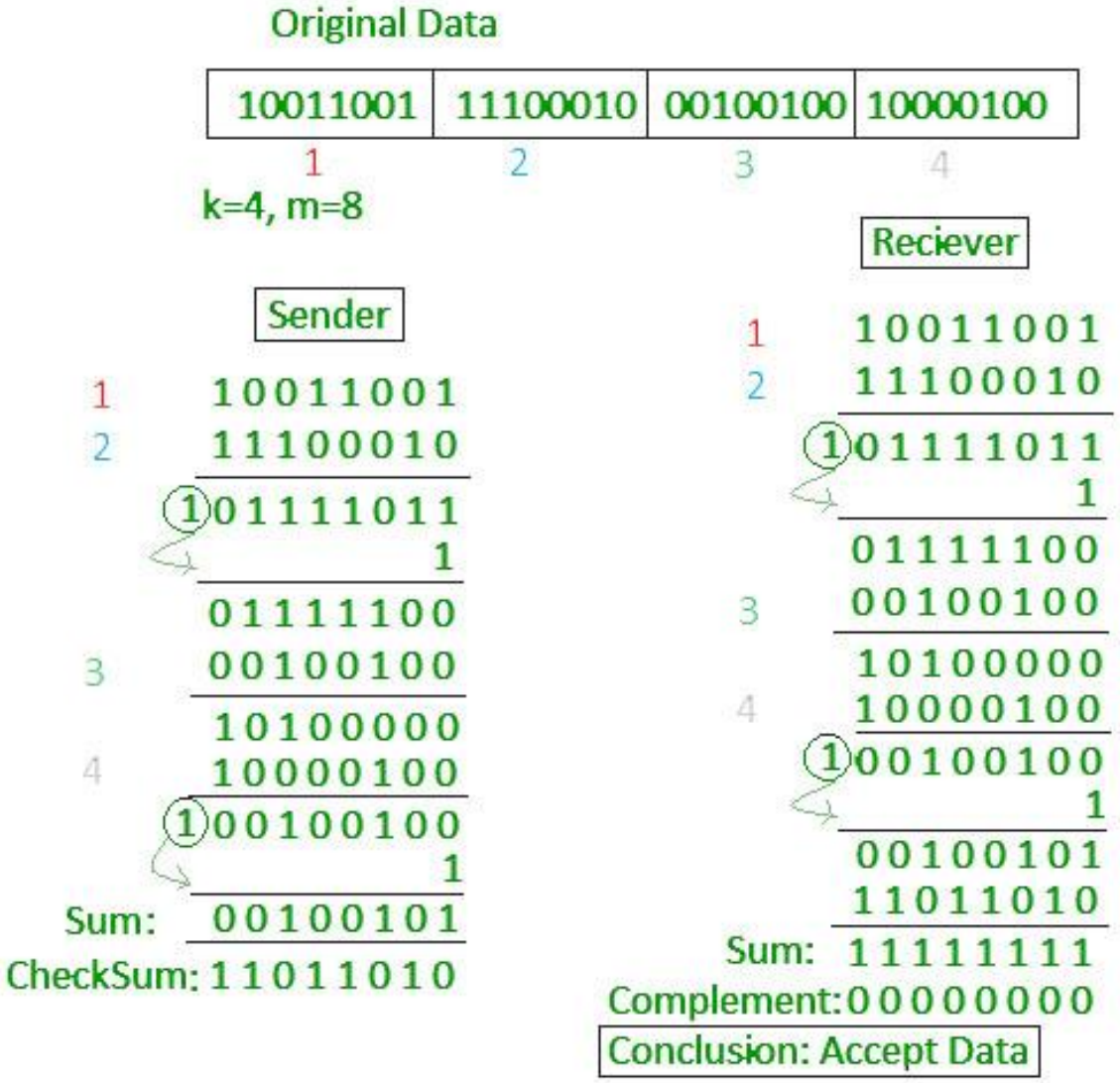


parity bit = **1** (in case of even parity setting)
parity bit = **0** (in case of odd parity setting)

Error Detection – Checksum

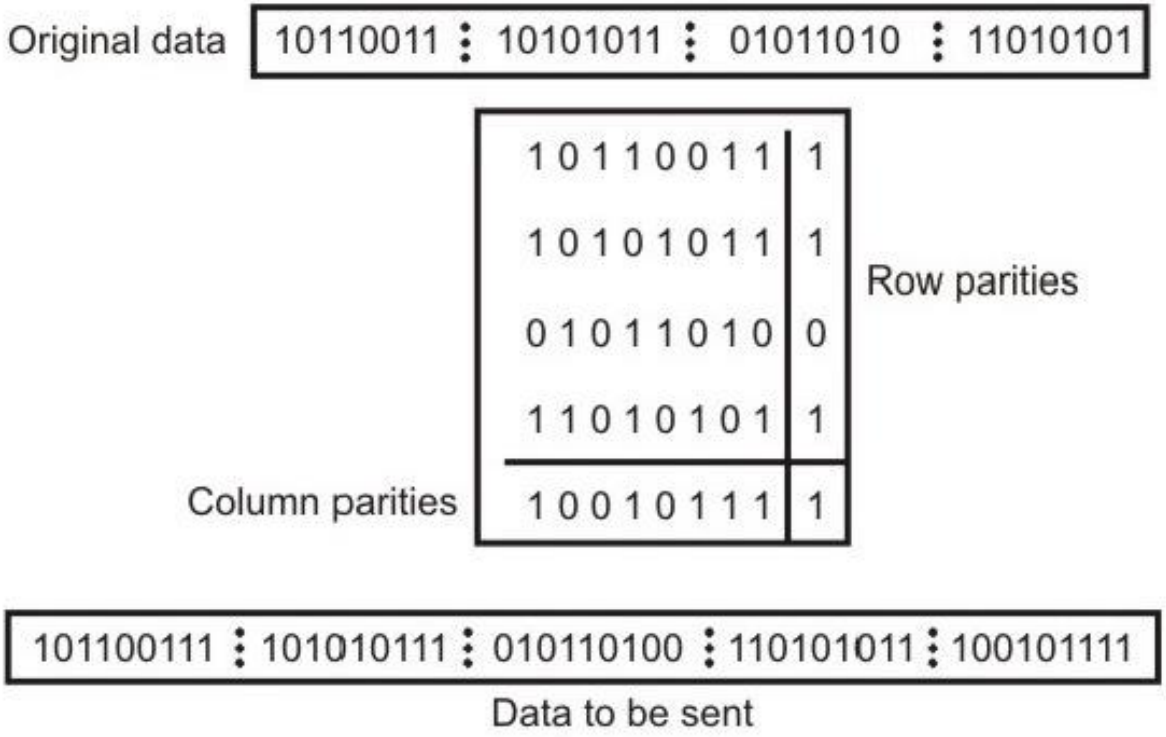
- A Checksum is an error detection technique based on the concept of redundancy.
- The process involves dividing the data into equally sized segments
- The segments are added using 1's complement arithmetic.
- The sum is complemented to get the checksum.
- The checksum segment is sent along with the data segments.
- At the receiver's end, all received segments are added using 1's complement arithmetic. The sum is complemented.
- If the result is zero, the data is correct.

Error Detection – Checksum



Error Correction – Two Dimensional Parity Check

- Two Dimensional Parity can detect as well as correct bit errors
- Make a Two Dimensional array from the message bit. After that, according to the even or odd parity, make the last column and last row of the matrix.



Error Correction – Two Dimensional Parity Check

two-dimensional bit parity:

- detect and correct single bit errors